

# Randomized Oversampling for Generalized Multiscale Finite Element Methods

**Victor M. Calo<sup>1,2</sup>, Yalchin Efendiev<sup>1,3\*</sup>, Juan Galvis<sup>4</sup>, Guanglian Li<sup>3</sup>**

<sup>1</sup> *Center for Numerical Porous Media (NumPor)  
King Abdullah University of Science and Technology (KAUST)  
Thuwal 23955-6900, Kingdom of Saudi Arabia.*

<sup>2</sup> *Applied Mathematics & Computational Science and Earth Science & Engineering  
King Abdullah University of Science and Technology (KAUST)  
Thuwal 23955-6900, Kingdom of Saudi Arabia.*

<sup>3</sup> *Department of Mathematics & Institute for Scientific Computation (ISC)  
Texas A&M University  
College Station, Texas, USA*

<sup>3</sup> *Departamento de Matemáticas  
Universidad Nacional de Colombia  
Bogotá D.C., Colombia*

---

## Abstract

In this paper, we study the development of efficient multiscale methods for flows in heterogeneous media. Our approach uses the Generalized Multiscale Finite Element (GMsFEM) framework. The main idea of GMsFEM is to approximate the solution space locally using a few multiscale basis functions. This is typically achieved by selecting an appropriate snapshot space and a local spectral decomposition, e.g., the use of oversampled regions in order to achieve an efficient model reduction. However, the successful construction of snapshot spaces may be costly if too many local problems need to be solved in order to obtain these spaces. In this paper, we show that this efficiency can be achieved using a moderate quantity of local solutions (or snapshot vectors) with random boundary conditions on oversampled regions with zero forcing. Motivated by the randomized algorithm presented in [19], we consider a snapshot space which consists of harmonic extensions of random boundary conditions defined in a domain larger than the target region. Furthermore, we perform an eigenvalue decomposition in this small space. We study the application of randomized sampling for GMsFEM in conjunction with adaptivity, where local multiscale spaces are adaptively enriched. Convergence analysis is provided. We present representative numerical results to validate the method proposed.

---

\*Email address: efendiev@math.tamu.edu

*Keywords:* Generalized multiscale finite element method, oversampling, high-contrast, randomized approximation, snapshot spaces construction.

---

## 1. Introduction

Model reduction is becoming increasingly important when dealing efficiently with problems characterized by multiple scales. Due to scale disparity, single-scale discretization techniques cannot provide useful results with acceptable computational cost in practice. In order to efficiently handle these multiscale problems, many model reduction techniques have been developed in the literature. These include approaches that are based on homogenization and numerical homogenization [6, 13, 18, 20], the approaches that employ finite element basis functions to approximate the fine-scale features of the solution space [1–3, 12, 17], and the approaches that employ global model reduction techniques [5, 7, 15]. In this paper, our focus is on approaches that are based on multiscale finite element methods which fall in the second category just mentioned. We use a recently introduced framework known as the Generalized Multiscale Finite Element Method (GMsFEM) and discuss how one can reduce the setup cost employing randomized Singular Value Decomposition (SVD) concepts [16, 19].

To construct multiscale basis functions, we employ the GMsFEM framework where the multiscale basis functions are constructed via a local spectral decomposition of a snapshot space. This snapshot space typically consists of spatial fields that represent the solution space up to some desired accuracy. For example, one choice for the snapshot space is to use harmonic functions that can represent any boundary value in each coarse region. These snapshots are constructed by solving local problems for all possible boundary conditions. The latter allows us to incorporate the effects of many small-scale features into these snapshots and thus achieve low dimensional coarse models. However, the computation of these snapshots is expensive. In this paper, we propose the use of random boundary conditions in constructing snapshot vectors. We show that by using only a few of these randomly generated snapshots, we can adequately approximate dominant modes of the solution space. To avoid oscillations near the boundary, the oversampling technique is used. More precisely, we solve local problems in domains that are larger than the target coarse blocks. Typically, they are larger by several layers of fine-grid blocks around the target coarse block. Furthermore, we perform a local spectral decomposition using the restriction of the randomly generated snapshots to the target coarse-grid domain.

The use of random boundary conditions (to generate the snapshot spaces) is motivated by the randomized SVD methodology [16, 19]. In general, randomized SVD algorithms allow computing dominant eigenvectors by considering a random linear combination of the columns (or rows) of a

given matrix. The random linear combinations typically have a component in the dominant modes and thus, by performing a spectral decomposition in the span of these random combinations, we can achieve an accurate approximation of dominant eigenvectors.

We take advantage of the idea of randomized linear combinations to considerably reduce the computational cost associated with the computation of snapshot vectors. In particular, we propose solving local problems with random boundary conditions and perform the local spectral decomposition in the space of these snapshots. The cost reduction is due to the fact that, in previous approaches, the snapshot spaces were constructed by solving local problems for every possible boundary condition in each coarse region. Using our new methodology, the number of snapshots to be generated is only slightly larger than the number of desired eigenvectors. Our experience suggests that for GMsFEM modeling, in general it suffices to include four additional random boundary conditions to the number of eigenvectors sought. For instance, in our numerical experiments, when three basis functions per coarse grid are needed, we compute only seven snapshot vectors (i.e., only seven random boundary conditions are generated). We discuss how the number of additional snapshots can depend on the eigenvalue structure for some special cases. This new methodology can provide substantial computational savings in the offline stage as we compute much fewer snapshots. We show that one needs to use randomized boundary conditions on the oversampled region to avoid oscillations near the boundaries. Indeed, if random boundary conditions are imposed on the target coarse grid (and no oversampling is used), the computed solution has oscillations near the boundaries which can cause large errors. Moreover, oversampling snapshots have several additional advantages [9] as they allow faster convergence for GMsFEM discretizations.

We compare the results obtained by using randomized snapshots to these obtained when all snapshot vectors are used. In the latter, we employ all possible boundary conditions on the oversampled region to construct the snapshot vectors. The local spectral decomposition is based on local eigenvalue problems, following previous studies [9]. Our numerical results show that one can achieve similar accuracy when using fewer random snapshots instead of using all possible snapshot vectors. Furthermore, we discuss approaches that can improve the results obtained by using randomized snapshots; however, at an additional computational cost.

We analyze the proposed method using [19, Lemma 18] and the convergence of oversampling GMsFEM [9]. In a first step, we estimate the approximation error between the full snapshots and randomized snapshots in each coarse neighborhood in a certain norm. This approximation error is used within GMsFEM analysis to show the convergence of the solution solved in the randomized snapshot space. We also discuss adaptive strategies for randomized snapshots. In adaptive methods, additional multiscale basis functions are added based on error estimators. These

estimators are proposed and investigated in [4]. Later in the paper, we discuss how additional multiscale basis functions can be computed by considering only a few extra random snapshots. In particular, in simulations we only compute four additional snapshot vectors in order to compute each additional multiscale basis function to be added as a refinement in the coarse domains that contain most error. The main objective of this paper is to show that the local snapshot spaces can be constructed inexpensively with an accuracy comparable to the state-of-the-art alternatives.

The paper is organized as follows. In Section 2, we give an introductory description of GMs-FEM. In Section 3, we present the randomized snapshot algorithm. Section 4 is devoted to numerical results. In this section, we also discuss the use of adaptive strategies and how to compute additional multiscale basis functions. In Section 5, we present the mathematical analysis of the method and in Section 6 we draw conclusions.

## 2. Preliminaries

We consider linear elliptic equations of the form

$$-\operatorname{div}(\kappa(x) \nabla u) = f \text{ in } D, \quad (1)$$

where  $u$  is prescribed on  $\partial D$ . We assume that the coefficient  $\kappa(x)$  has multiple scales and high variations (e.g., see Fig. 1). In this paper we focus on the two dimensional case but our methodology can be easily extended to problems in three dimensions, where the implied savings could be larger.

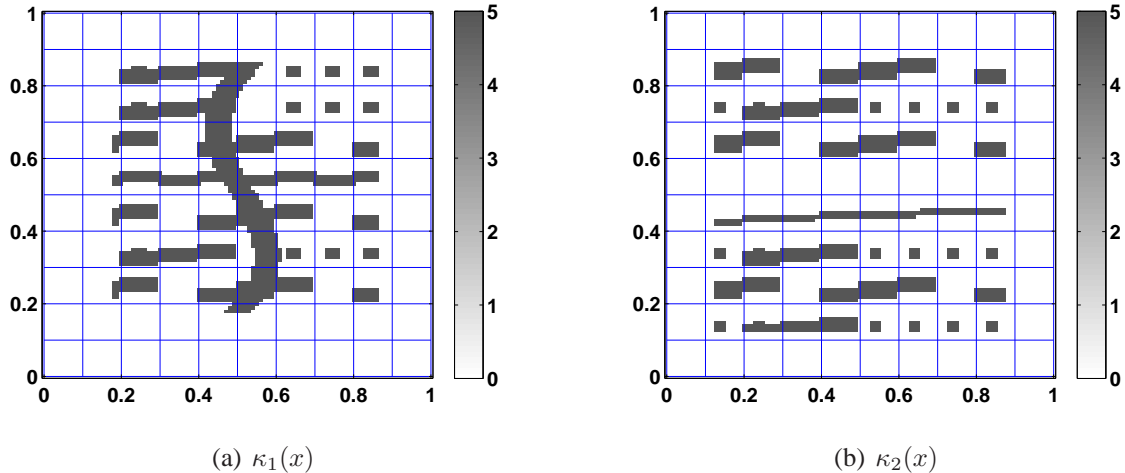


Figure 1: Permeability fields in  $\log_{10}$ -scale.

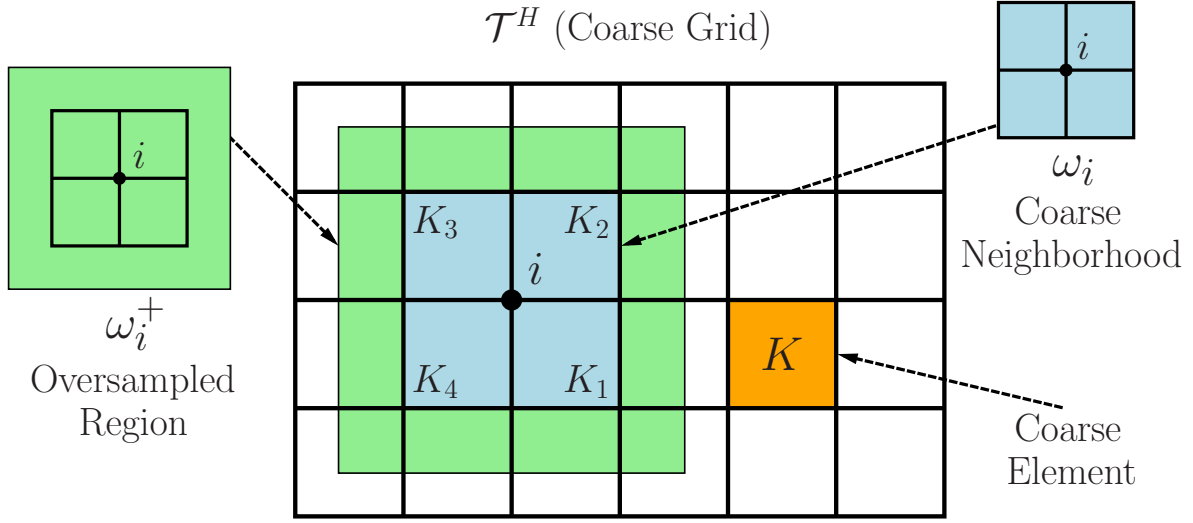


Figure 2: Illustration of a coarse neighborhood and oversampled domain. Here,  $K$  is a coarse-grid block,  $\omega_i$  is a coarse neighborhood of  $x_i$ , and  $\omega_i^+$  is an oversampled region

### 2.1. Fine and coarse grids

Let  $\mathcal{T}^H$  be a conforming partition of the computational domain  $D$  into finite elements denoted by  $\{K_j\}$  (triangles, quadrilaterals, tetrahedra, etc.), called coarse grid. Assume that each coarse subregion is partitioned into a connected union of fine-grid blocks. Assume the fine grids match across coarse elements boundaries and denote by  $\mathcal{T}^h$  the obtained (fine-grid) triangulation of  $D$ . We use  $\{x_i\}_{i=1}^{N_c}$  (where  $N_c$  the number of coarse nodes) to denote the vertices of the coarse mesh  $\mathcal{T}^H$ , and define the neighborhood of the node  $x_i$  by

$$\omega_i = \bigcup \{K_j \in \mathcal{T}^H; \quad x_i \in \overline{K_j}\}. \quad (2)$$

See Fig. 2 for an illustration of neighborhoods and elements subordinated to the coarse discretization. We introduce notation for oversampled regions. We denote by  $\omega_i^+$  the oversampled region of  $\omega_i \subset \omega_i^+$ , defined by adding several fine- or coarse-grid layers around  $\omega_i$ . We emphasize that the coarse-grid is too coarse to effectively resolve all heterogeneities and scales present in the coefficient  $\kappa$ , while the fine grid resolves all variations of  $\kappa$  but it leads to a huge linear system that is not practical to solve.

## 2.2. Generalized Multiscale Finite Element Method (GMsFEM)

Throughout this paper, we use the continuous Galerkin formulation, and use  $\omega_i$  as the support of basis functions. The regions  $\omega_i^+$  are used to construct the multiscale basis functions. For the purpose of this description, we formally denote the basis functions of the offline space  $V_{\text{off}}$  by  $\phi_k^{\omega_i}$ . The solution is sought as  $u_H(x) = \sum_{i,k} c_k^i \phi_k^{\omega_i}(x)$ , where  $k$  denotes the basis function index in the domain  $\omega_i$ . Once the basis functions are identified, we solve

$$a(u_H, v) = (f, v), \quad \text{for all } v \in V_{\text{off}}, \quad (3)$$

and

$$a(u, v) = \int_D \kappa(x) \nabla u \cdot \nabla v.$$

Now, we briefly describe GMsFEM. We consider oversampling for GMsFEM (see [8, 9]) that uses harmonic snapshots. That is, snapshots vector are obtained as harmonic extensions of some subset of all possible boundary conditions on the oversampled domain. We construct a snapshot space  $V_{\text{snap}}^{\omega_i^+}$ . Construction of the snapshot space involves solving local problems and we detail the standard process below [8, 9].

The snapshot space consists of harmonic extensions of fine-grid functions defined on the boundary of  $\omega_i^+$ . More precisely, for each fine-scale function with support on the boundary of the oversampled coarse domain,  $\delta_l^h(x)$ , we solve a local problem. Let  $\delta_l^h(x_k) = \delta_{lk}$  be one of these functions where for all  $l, k \in J_h(\omega_i^+)$ , where  $J_h(\omega_i^+)$  is the fine-grid boundary nodes on  $\partial\omega_i^+$  and  $\delta_{lk}$  is Kronecker's delta with value 1 for  $k = l$  and value 0 otherwise. Thus, the local problem to solve is

$$-\text{div}(\kappa(x) \nabla \psi_{l,\omega_i}^{+, \text{snap}}) = 0 \quad \text{in } \omega_i^+ \quad (4)$$

subject to boundary condition,  $\psi_{l,\omega_i}^{+, \text{snap}} = \delta_l^h(x)$  on  $\partial\omega_i^+$ . We form the snapshot matrices by placing the solutions of these local problems as the rows of this matrix (throughout, for notational convenience, we do not distinguish between the fine-grid vectors and their continuous representations)

$$\Psi_{\omega_i}^{+, \text{snap}} = [\psi_{1,\omega_i}^{+, \text{snap}}, \dots; \psi_{l,\omega_i}^{+, \text{snap}}, \dots].$$

We define the vectors  $\psi_{l,\omega_i}^{\text{snap}}$  as the restrictions of the snapshot vectors  $\psi_{l,\omega_i}^{+, \text{snap}}$  to degrees of freedom in  $\omega_i$  by taking their values at the fine-grid nodes of  $\omega_i$ . Considering these vectors, we form the snapshot matrix in  $\omega_i$

$$\Psi_{\omega_i}^{\text{snap}} = [\psi_{1,\omega_i}^{\text{snap}}, \dots; \psi_{l,\omega_i}^{\text{snap}}, \dots]. \quad (5)$$

Next, we discuss the construction of a smaller offline space using an eigenvalue problem [8].

In order to construct an offline space  $V_{\text{off}}$ , we reduce the dimension of the snapshot space using an auxiliary spectral decomposition. We seek a subspace of the snapshot space where to approximate any element of the snapshot space in the appropriate norm defined via the following auxiliary bilinear forms. For each  $\omega_i$ , we define

$$A^{\text{off}} \Theta_k^{\text{off}} = \lambda_k^{\text{off}} S^{\text{off}} \Theta_k^{\text{off}}, \quad (6)$$

where

$$A^{\text{off}} = [a_{mn}^{\text{off}}] = \int_{\omega_i} \kappa(x) \nabla \psi_{m,\omega_i}^{+, \text{snap}} \cdot \nabla \psi_{n,\omega_i}^{+, \text{snap}} = \Psi_{\omega_i}^{\text{snap}} A (\Psi_{\omega_i}^{\text{snap}})^T$$

and

$$S^{\text{off}} = [s_{mn}^{\text{off}}] = \int_{\omega_i} \tilde{\kappa}(x) \psi_{m,\omega_i}^{+, \text{snap}} \psi_{n,\omega_i}^{+, \text{snap}} = \Psi_{\omega_i}^{\text{snap}} S (\Psi_{\omega_i}^{\text{snap}})^T.$$

The coefficient  $\tilde{\kappa}(x)$  uses multiscale partition of unity functions (cf., [8]) which is described in (9). Here,  $A$  and  $S$  are fine-grid stiffness and mass matrices in the coarse region. To generate the offline space, we then choose the smallest  $M_{\text{off}}$  eigenvalues of Eqn. (6) for each  $\omega_i^+$  and form the corresponding eigenvectors in the respective space of snapshots by setting  $\psi_{k,\omega_i}^{+, \text{off}} = \sum_j \Theta_{kj}^{\text{off}} \psi_{j,\omega_i}^{+, \text{snap}}$  (for  $k = 1, \dots, M_{\text{off}}$ ), where  $\Theta_{kj}^{\text{off}}$  are the components of the vector  $\Theta_k^{\text{off}}$ . We then create the offline matrices

$$\Psi_{\omega_i}^{+, \text{off}} = [\psi_{1,\omega_i}^{+, \text{off}}, \dots, \psi_{\omega_i, M_{\text{off}}}^{+, \text{off}}] \quad \text{and} \quad \Psi_{\omega_i}^{\text{off}} = [\psi_{1,\omega_i}^{\text{off}}, \dots, \psi_{M_{\text{off}}, \omega_i}^{\text{off}}],$$

where  $\psi_{k,\omega_i}^{\text{off}}$  is the restriction of  $\psi_{k,\omega_i}^{+, \text{off}}$  to  $\omega_i$ . To construct multiscale basis functions, we multiply the dominant eigenvectors by a partition of unity functions  $\chi_i$  that are supported in  $\omega_i$ , such that  $\sum_i \chi_i = 1$ . More precisely, the offline space is composed of the following basis functions,

$$\phi_k^{\omega_i} = \chi_i \psi_k^{\omega_i}. \quad (7)$$

We can choose the partition of unity functions to be multiscale finite element basis functions; see [11]. Let  $\chi_i^0$  be the nodal basis of the standard finite element space  $W_H$ . For example,  $W_H$  consists of piecewise linear functions if  $\mathcal{T}_H$  is a triangular partition or  $W_H$  consists of piecewise bilinear functions if  $\mathcal{T}_H$  is a rectangular partition. “Standard” multiscale finite element basis functions coincide with  $\chi_i^0$  on the boundaries of the coarse partition and satisfy:

$$\text{div}(\kappa \nabla \chi_i^{ms}) = 0 \text{ in } K \in \omega_i, \quad \chi_i^{ms} = \chi_i^0 \text{ in } \partial K, \quad \text{for all } K \in \omega_i, \quad (8)$$

where  $K$  is a coarse grid block within  $\omega_i$ . In our numerical implementations, we take  $\tilde{\kappa} = \kappa$  for

the computation of mass matrix. However, one can take a weighted permeability field (see detailed discussion in [8]) such as

$$\tilde{\kappa} = \sum_i \kappa |\nabla \chi_i^+|^2. \quad (9)$$

### 3. Randomized Oversampling

As described above, a usual choice for the snapshot space consists of the harmonic extension of fine-grid functions defined on the boundary of  $\omega_i^+$ . This type of snapshot is complete in the sense that it captures all the boundary information of the solution. However, the computational cost is expensive since, in each local coarse neighborhood,  $O(n^{\omega_i^+})$  number of local problems is required to solve. Here,  $n^{\omega_i^+}$  denotes the number of fine grids on the boundary of  $\omega_i^+$ . A smaller yet accurate snapshot space is needed to build a more efficient multiscale method.

In the following, we generate inexpensive snapshots using random boundary conditions. That is, instead of solving Eqn. (4) for each fine boundary node, we solve a small number of local problems imposed with random boundary conditions:

$$\psi_{l,\omega_i}^{+, \text{rsnap}} = r_l \text{ on } \partial\omega_i^+, \quad (10)$$

where  $r_l$  are independent identically distributed (i.i.d.) standard Gaussian random vectors on the fine-grid nodes of the boundary. Then, we can obtain the local random snapshot on the target domain  $\omega_i$  by restricting the solution of this local problem,  $\psi_{l,\omega_i}^{+, \text{rsnap}}$  to  $\omega_i$  (which is denoted by  $\psi_{l,\omega_i}^{\text{rsnap}}$ ). The space generated by  $\psi_{l,\omega_i}^{\text{rsnap}}$  is a subspace of the space generated by all local snapshots  $\Psi_{\omega_i}^{\text{snap}}$ . Therefore, there exists a randomized matrix  $\mathcal{R}$  with rows composed by the random boundary vectors  $r_l$ , such that,

$$\Psi_{\omega_i}^{\text{rsnap}} = \mathcal{R} \Psi_{\omega_i}^{\text{snap}}. \quad (11)$$

Using these snapshots, we follow the procedure in the previous section to generate multiscale basis functions. Below, we summarize the algorithm. We denote the buffer number  $p_{\text{bf}}^{\omega_i}$  for each  $\omega_i$  and the number of local basis functions by  $k_{\text{nb}}^{\omega_i}$  for each  $\omega_i$ . Later on, we use the same buffer number for all  $\omega_i$  and simply use the notation  $p_{\text{bf}}$ .

### 4. Numerical results

In this section, we present representative numerical experiments that demonstrate the good performance of the randomized snapshots algorithm. We take the domain  $D$  as a square, set the



Table 1: Randomized GMsFEM Algorithm

---



---

<b>Input:</b>	Fine grid size $h$ , coarse grid size $H$ , oversampling size $t$ , buffer number $p_{\text{bf}}^{\omega_i}$ for each $\omega_i$ , the number of local basis functions $k_{\text{nb}}^{\omega_i}$ for each $\omega_i$ ;
<b>output:</b>	Coarse-scale solution $u_H$ .
	1. Generate oversampling region for each coarse block: $\mathcal{T}^H$ , $\mathcal{T}^h$ , and $\omega_i^+$ ;
	2. Generate $k_{\text{nb}}^{\omega_i} + p_{\text{bf}}^{\omega_i}$ random vectors $r_l$ and obtain randomized snapshots in $\omega_i^+$ (Eqn. (10)); Add a snapshot that represents the constant function on $\omega_i^+$ ;
	3. Obtain $k_{\text{nb}}^{\omega_i}$ offline basis by a spectral decomposition (Eqn. (6) restricted to random snapshots);
	4. Construct multiscale basis functions (Eqn. (7)) and solve (Eqn. (3)).

---



---

forcing term  $f = 0$  and use a linear boundary condition for the problem (1), that is,  $u = x_1 + x_2$  on  $\partial D$  where  $x_i$  are the Cartesian components of each point. In our numerical simulations, we use a coarse grid of  $10 \times 10$  blocks, and each coarse grid block is divided into  $10 \times 10$  fine grid blocks. Thus, the whole computational domain is partitioned by a  $100 \times 100$  fine grid. We use a few multiscale basis functions per coarse block. These coarse basis set defines the problem size. We assume that the fine-scale solution is obtained by discretizing problem (1) by the classical conforming piecewise bilinear elements on the fine grid. To test the performance of our algorithm, we consider two permeability fields  $\kappa$  as depicted in Figure 1. The first permeability field (left figure) has more connected regions and they are more irregular compared to the second permeability field (right figure). We observed similar behavior for these two cases, and therefore we focus on the numerical results for the first permeability field (Figure 1(a)).

In Table 2, a comparison between using all snapshots and the randomized snapshots is shown. The first column shows the dimension of the offline space for each test. We choose 5, 10, 15, 20, and 25 basis functions per each interior node (in addition to the constant eigenvectors) and use an oversampling layer that consists of three fine-grid blocks ( $t = 3$ ). The offline space  $V_{\text{off}}$  is defined via a local spectral decomposition as specified in Section 3. The snapshot ratio is calculated as the number of randomized snapshots divided by the number of the full snapshots. This ratio is displayed in the second column. Here, the total number of snapshots refers to the number of boundary nodes of the oversampled region. In our numerical results, an oversampled region has  $26 \times 26$  fine-grid dimension and there are total 104 snapshots if all boundary nodes are used. For example, when the dimension of the offline space is 931, we only compute 14 snapshots instead of 104. This ratio gives the information on the computational savings of our algorithm compared to the previous algorithm using all snapshots. The next two columns shows the relative weighted  $L^2$  error and relative energy error using the full snapshots. The weighted  $L^2$  norm and energy norm

are defined as

$$\|u\|_{L^2_\kappa} = \left( \int_D \kappa u^2 \right)^{\frac{1}{2}} \quad \text{and} \quad \|u\|_{H^1_\kappa} = \left( \int_D \kappa |\nabla u|^2 \right)^{\frac{1}{2}},$$

respectively. Further, the relative weighted  $L^2$  error and relative energy error using the randomized snapshots are shown in the last two columns. From this table, we observe that the randomized algorithm converges in the sense that the relative error decreases as we increase the dimension of the coarse space. Comparing the fourth column with the last column, we conclude that the accuracy when using the randomized snapshots is similar to using all snapshot vectors. The latter has much larger dimension as shown in the second column that shows the percentage of the snapshots computed. Therefore, the proposed method is an order of magnitude faster while having comparable accuracy. For example, when the dimension of the offline space is 931, the accuracy of the methods is comparable while randomized snapshot approach uses only 13.46% of the snapshots. Similar results are obtained when the fine mesh is refined to  $200 \times 200$ . In particular, with the offline space with the dimension 931 and the snapshot ratio of 10%, we obtain similar  $L^2_\kappa(D)$  and  $H^1_\kappa(D)$  errors which are 1.28% and 24.02%. The behavior is similar when we use the permeability field in Fig. 1(b). The results are displayed in Table 3. Here,  $p_{\text{bf}}$  refers to the buffer that is used to compute the eigenvectors. For example,  $p_{\text{bf}} = 4$  means that we use  $n + 4$  snapshots to compute  $n$  basis functions for each coarse block.

Table 2: Numerical results comparing the results between using all harmonic snapshots and the snapshots generated by random boundary conditions with  $p_{\text{bf}} = 4$ ,  $\kappa$  as shown in Fig. 1(a). In the parenthesis, we show a higher value of the snapshot ratio.

dim( $V_{\text{off}}$ )	Snapshot ratio (%)	All snapshots (%)		Few randomized snapshots (%)	
		$L^2_\kappa(D)$	$H^1_\kappa(D)$	$L^2_\kappa(D)$	$H^1_\kappa(D)$
526	8.65(15.38)	0.87	18.15	2.81(1.38)	44.95(26.04)
931	13.46	0.64	14.85	1.04	23.61
1336	18.27	0.55	13.59	0.70	18.08
1741	23.08	0.50	12.69	0.64	15.91
2146	27.88	—	—	0.54	14.16

In Fig. 3, the fine-scale solution, coarse-scale solution using all snapshots and coarse-scale solution using randomized snapshots are shown. They are obtained using the second test (when the dimension of the offline space is 931) in Table 2. These two coarse-scale solutions are a good approximation of the fine-scale solution. This is corroborated in Fig. 4, where we plot the absolute error of the two solutions.

Next, we investigate the effect of the buffer number  $p_{\text{bf}}$  on the accuracy of the coarse solution. We test a series of simulations with different  $p_{\text{bf}}$  while keeping the coefficients and meshes fixed.

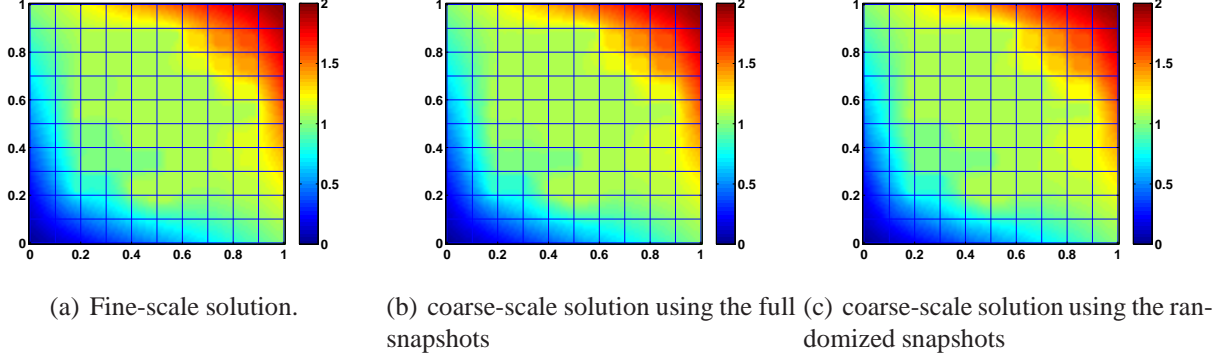


Figure 3: The fine-scale solution and coarse-scale solutions correspond to Fig. 1(a).

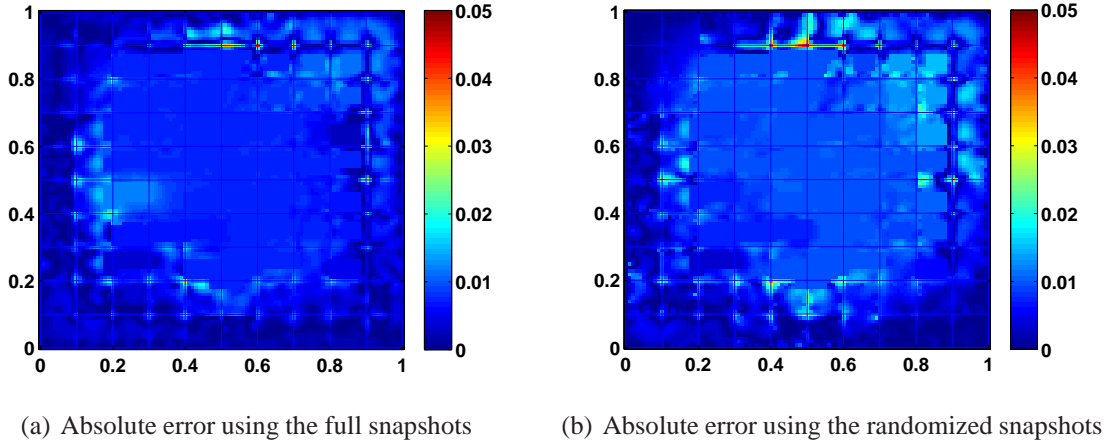


Figure 4: The absolute errors correspond to Fig. 1(a) using full snapshots and random snapshots.

Table 3: Numerical results comparing the results between using all harmonic snapshots and the snapshots generated by random boundary conditions with  $p_{\text{bf}} = 4$ ,  $\kappa$  as shown in Fig. 1(b).

$\dim(V_{\text{off}})$	snapshot ratio (%)	all snapshots (%)		using the randomized snapshots (%)	
		$L^2_{\kappa}(D)$	$H^1_{\kappa}(D)$	$L^2_{\kappa}(D)$	$H^1_{\kappa}(D)$
526	8.65(15.38)	0.71	20.98	1.33(0.80)	33.76(24.14)
931	13.46	0.51	17.33	0.66	21.67
1336	18.27	0.45	15.83	0.53	18.26
1741	23.08	0.40	14.66	0.48	17.13
2146	23.88	—	—	0.43	15.39

The results are presented in Table 4, which shows that a larger buffer coefficient decreases the relative energy error. However, there is no need for very large values. If we take  $p_{\text{bf}} = 4$ , we can get a coarse solution with error of 15.51%, while obtaining a 14.49% error if using  $p_{\text{bf}} = 20$  at the cost of solving 16 extra local problems for each inner coarse node.

Table 4: Numerical results for different  $p_{\text{bf}}$  and using 20 local basis in each coarse neighborhood,  $\kappa$  as shown in Fig. 1(a).

$p_{\text{bf}}$	$\ u - u^{\text{off}}\ $ (%)	
	$L_{\kappa}^2(D)$	$H_{\kappa}^1(D)$
4	0.62	15.51
10	0.62	15.08
15	0.57	14.70
20	0.57	14.49

Table 5: Numerical results for different oversampling domain  $\omega_i^+ = \omega_i + t$  and using 20 local basis in each coarse neighborhood,  $p_{\text{bf}} = 4$ ,  $\kappa$  as shown in Fig. 1(a).

$t$	$\ u - u^{\text{off}}\ $ (%)	
	$L_{\kappa}^2(D)$	$H_{\kappa}^1(D)$
0	1.52	23.26
2	0.61	15.63
4	0.62	15.56
7	0.59	15.24

Lastly, numerical tests are conducted to study the influence of oversampling effects on the accuracy of the randomized snapshots. The simulation results are shown in Table 5. From this table, we observe that oversampling technique is needed to obtain an accurate solution. However, a larger oversampling domain is not necessary since it increases the computational cost of the solution, while no significant improvement in the solution accuracy is observed.

#### 4.1. Comparison of results of different spectral problems

As we mentioned in the introduction, Section 1, one can use solution-based boundary conditions to achieve higher accuracy compared to the random boundary conditions. In this section, we demonstrate this. The main idea behind this algorithm is to select boundary modes using a small spectral decomposition over the boundary layer  $\mathcal{L}_i$  instead of the oversampling region  $\omega_i^+$  that surrounds the boundary in the spectral problem Eqn. (6). More precisely, we consider a local spectral problem in the layer of a few fine-grid blocks in the region that contains the boundary of  $\omega_i$  (see Fig. 5). We choose a layer that has a thickness of five fine-grid elements (two interior to  $\omega_i$  and three on the immediate neighborhood of  $\omega_i$ ). Furthermore, we select dominant eigenvectors (corresponding to smallest eigenvalues) by solving local eigenvalue problem in the strip. The local eigenvalue problem uses local stiffness and mass matrices (as in [10, 14]). This approach provides correct fine-scale features and we expect higher accuracy compared to the randomized snapshots. The numerical results are shown in Table 6. Comparing the fourth column with the last column of

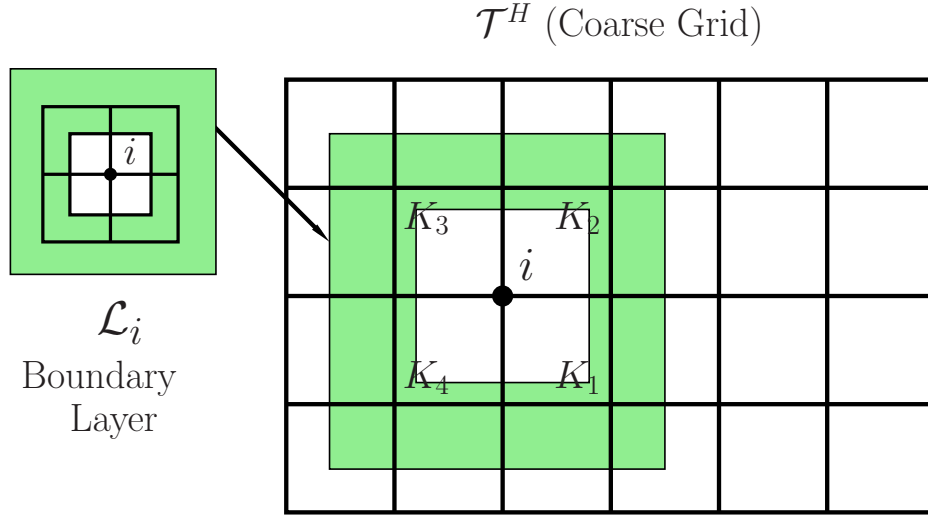


Figure 5: Illustration of a skin layer  $\mathcal{L}_i$  that is used for computing boundary conditions for the snapshots in  $\omega$ .

Table 6, we observe that this new algorithm is more accurate compared to the previous one. Taking the fifth row as an example, for the same dimension of the offline space, the new algorithm gives 14.97% error while the previous algorithm ends with 17.13%. In general, one can apply randomized snapshot algorithms to reduce the computational cost associated with our new algorithm. That is, one can use randomized snapshots for the strip  $\mathcal{L}_i$  to reduce the computational cost further.

Table 6: Numerical results comparing the results between the snapshots obtained from skin layer spectral problems and the snapshots generated by random boundaries with  $p_{\text{bf}} = 4$ ,  $\kappa$  as shown in Fig. 1(b).

dim( $V_{\text{off}}$ )	snapshot ratio (%)	snapshots from skin layer (%)		randomized snapshots (%)	
		$L_{\kappa}^2(D)$	$H_{\kappa}^1(D)$	$L_{\kappa}^2(D)$	$H_{\kappa}^1(D)$
526	8.65	1.03	26.51	1.33	33.76
931	13.46	0.63	18.64	0.66	21.67
1336	18.27	0.48	16.29	0.53	18.26
1741	23.08	0.42	14.97	0.48	17.13
2146	27.88	0.39	14.40	0.43	15.39

#### 4.2. A randomized multiscale adaptive algorithm

In this section, we discuss how to efficiently use randomized snapshots within adaptive algorithms. We use the error indicators developed in [4]. First, we briefly recall these error estimators.

Let  $V_i = H_0^1(\omega_i)$ , define a linear functional  $R_i(v)$  on  $V_i$  by

$$R_i(v) = \int_{\omega_i} f v - \int_{\omega_i} a \nabla u_{\text{ms}} \cdot \nabla v, \quad (12)$$

where the norm of  $R_i$  is defined as

$$\|R_i\|_{V_i^*} = \sup_{v \in V_i} \frac{|R_i(v)|}{\|v\|_{V_i}}. \quad (13)$$

Here  $\|v\|_{V_i} = (\int_{\omega_i} \kappa(x) |\nabla v|^2 dx)^{\frac{1}{2}}$ . In [4] it is shown that

$$\|u - u_H\|_V^2 \leq C_{\text{err}} \sum_{i=1}^N \|R_i\|_{V_i^*}^2 (\lambda_{l_i+1}^{\omega_i})^{-1}, \quad (14)$$

where  $C_{\text{err}}$  is a uniform constant and  $\lambda_{l_i+1}^{\omega_i}$  denotes the  $(l_i + 1)$ -th eigenvalue over coarse neighborhood  $\omega_i$  that corresponds to the first eigenvector excluded from the construction of  $V_{\text{off}}$ . We define the error indicator in each coarse neighborhood as follows,

$$\eta_i^2 = \|R_i\|_{V_i^*}^2 (\lambda_{l_i^m+1}^{\omega_i})^{-1}, \quad \text{for } H^{-1}\text{-based residual.}$$

The pivotal issue to solve is to generate additional linearly independent basis for a selected coarse neighborhood  $\omega_i$  for the current iteration. Specifically, those extra basis are required to be linearly independent from the basis in the previous iteration. In what follows, we describe a possible solution to this issue using the residue of a series of random basis and their projection onto the offline space of the previous iteration.

**Remark 1.** *The Step 2 in Table 7 is to guarantee that the added local basis are independent from the previous local basis in the  $M$ -norm as defined in the next section. In the randomized snapshots, we have added the constant local basis manually to guarantee that the multiscale basis are included. However, this constant basis should be excluded in Step 2 since the constant is not in the spectral vectors and if it is added, we can get linear dependency.*

The numerical results are displayed in Table 8. First, we take five (5) basis per coarse node. Then, we apply the multiscale adaptive algorithm proposed in [4] and identify the coarse nodes index  $I$  requiring more basis. Set  $c_{nb}^{\omega_i} = 2$  and  $c_{bf}^{\omega_i} = 1$  and follow Table 7, next, we generate  $c_{nb}^{\omega_i} + c_{bf}^{\omega_i} = 3$  local random basis for those nodes and use Step 2 to get three new linearly independent basis. Afterwards, a local spectral decomposition is performed to select two important basis from

Table 7: Local basis enrichment algorithm

---



---

<b>Input:</b>	an index of the coarse nodes $I$ selected by the error indicator for enrichment, the local offline space $\Psi_{\omega_i}^{\text{rsnap}}$ , buffer number $c_{\text{bf}}^{\omega_i}$ , an additional local basis number $c_{\text{nb}}^{\omega_i}$ for each $i \in I$ .
<b>output:</b>	an enriched local offline space $\Psi_{\omega_i}^{\text{rsnap}}$ corresponds to each nodes in $I$ .
1.	Generate $c_{\text{nb}}^{\omega_i} + c_{\text{bf}}^{\omega_i}$ random vectors $r_l$ and obtain randomized snapshots in $\omega_i^+$ (Eqn. (10)). Denote as $\phi_1, \dots, \phi_{c_{\text{nb}}^{\omega_i} + c_{\text{bf}}^{\omega_i}}$ ;
2.	A modification of the random basis obtained from Step 1: $\tilde{\phi}_i = \phi_i - \sum_{j=1}^N \frac{\langle \phi_i, \psi_j \rangle_M}{\langle \psi_j, \psi_j \rangle_M} \phi_j$ , where $\psi_1, \dots, \psi_N$ denote a series of basis of $\Psi_{\omega_i}^{\text{rsnap}}$ excluded the constant one;
3.	Obtain $c_{\text{nb}}^{\omega_i}$ offline basis by a spectral decomposition (Eqn. (6)), next, add in a snapshot that represent the constant function on $\omega_i^+$ , and denote the resulting vectors as $\Psi_{\omega_i}^{\text{enrich}}$ ;
4.	$\Psi_{\omega_i}^{\text{rsnap}} \equiv \Psi_{\omega_i}^{\text{rsnap}} \cup \Psi_{\omega_i}^{\text{enrich}}$ .

---



---

those three basis. In the end, the corresponding multiscale basis functions are constructed and added to the coarse space.

Comparing Tables 3 and 8, we observe that the randomized adaptive algorithm is cheaper since much fewer basis functions are used to achieve comparable accuracy to that of the uniform increase of basis shown in Table 3. 2146 basis functions are calculated to attain an energy error of 14.16% in Table 3, while only 2061 are necessary to get a smaller error of 13.90% using the adaptive randomized algorithm. Here, we do not discuss the computational cost of our adaptive algorithm and refer to [4] for details. Our main goal in this section is simply to demonstrate how additional basis functions can be computed using a small set of new snapshots that avoids storing all the eigenvectors.

Table 8: Numerical results using adaptive algorithm with  $p_{\text{bf}} = 4$ , and 5 local basis per node at the beginning and with two more basis for selected nodes,  $\kappa$  as shown in Fig. 1(a).

dim( $V_{\text{off}}$ )	using the usual snapshots (%)	
	$L_{\kappa}^2(D)$	$H_{\kappa}^1(D)$
526	4.11	50.23
916	0.99	21.65
1323	0.63	17.33
1717	0.53	15.10
2061	0.51	13.90

## 5. Analysis

In the analysis described below, we first estimate the error due to the approximation using randomized snapshots. In the first lemma, we compare an arbitrary snapshot obtained using all snapshot vectors and its approximation in the space of randomized snapshots. To avoid cumbersome notation, we denote the local snapshot matrix  $\Psi_{\omega_i}^{\text{snap}}$  in (5) by  $\Psi$  and the local randomized snapshot matrix  $\Psi_{\omega_i}^{\text{rsnap}}$  in (11) by  $\Psi^r$ .

The following lemma shows that the randomized snapshot  $\Psi^r$  with  $l$  random basis is a good approximation of the full snapshot  $\Psi$  composed of  $m$  basis,  $m > l$ . We use the notation  $A \preceq B$  when  $A \leq CB$  with  $C$  being independent of the size ratio between the coarse and fine meshes, and spatial scales. Throughout,  $\|\cdot\|$  denotes the  $l^2$  norm for vectors and the  $l^2$ -based spectral norm for matrices, while  $\|z\|_A = z^T A z$ . We remind that, throughout, for notational convenience, we do not distinguish between the fine-grid vectors and their continuous representations.

**Lemma 2.** *Suppose  $\Psi \in \mathbf{R}^{m \times n}$  of rank  $m$ , and  $\mathcal{R} \in \mathbf{R}^{l \times m}$  whose entries are i.i.d. Gaussian random variables. Define  $\Psi^r = \mathcal{R}\Psi$ , then, for any  $\xi \in \mathbf{R}^m$ , there exists  $\xi^r \in \mathbf{R}^l$ , such that*

$$\|\xi^T \Psi - (\xi^r)^T \Psi^r\|_{\widetilde{M}(\omega_i)}^2 = \int_{\omega_i} \widetilde{\kappa} |\xi^T \Psi - (\xi^r)^T \Psi^r|^2 \preceq \left( \frac{\|\mathcal{H}^{(-1)} \mathcal{S}\| + 1}{\lambda_{k+1}} \right)^2 \|\xi^T \Psi\|_{A(\omega_i)}^2, \quad (15)$$

where  $k < l < m < n$ , and  $\mathcal{S}$ ,  $\mathcal{H}$ , and  $T$  are defined in Eqns. (17) and (19).

Here,  $\lambda_{k+1}$  is the  $(k+1)^{\text{th}}$  smallest diagonal value of  $\Lambda$  defined in Eqn. (16) and  $A(\omega_i) = (\int_{\omega_i} \kappa \nabla \phi_j \nabla \phi_k)_{n \times n}$  with  $\phi_j$  as the  $j^{\text{th}}$  local fine-scale basis in the  $\omega_i$ . Besides,  $\|\xi^T \Psi\|_{\widetilde{M}(\omega_i)} = (\int_{\omega_i} \widetilde{\kappa} \xi^T \Psi \xi^T \Psi)^{\frac{1}{2}}$ ,  $\|\xi^T \Psi\|_{A(\omega_i)} = (\int_{\omega_i} \kappa \nabla(\xi^T \Psi) \cdot \nabla(\xi^T \Psi))^{\frac{1}{2}}$ .

*Proof.* Denote  $\widetilde{M}(\omega_i) = (\int_{\omega_i} \widetilde{\kappa} \phi_j \phi_k)_{n \times n}$  with  $\phi_j$  as the  $j^{\text{th}}$  local fine-scale basis in  $\omega_i$ . The matrix  $\widetilde{M}(\omega_i)$  is symmetric positive definite. Besides,  $A(\omega_i)$  is symmetric semi positive definite. Thus, there exists an  $m \times m$  matrix  $U$ , such that

$$U^T \Psi \widetilde{M}(\omega_i) \Psi^T U = \Lambda, \text{ and } U^T \Psi A(\omega_i) \Psi^T U = I, \quad (16)$$

where  $I$  is an identity matrix and  $\Lambda$  denotes a diagonal matrix with decreasing diagonal values

$$\frac{1}{\lambda_1}, \frac{1}{\lambda_2}, \dots, \frac{1}{\lambda_m}.$$

Define  $X = U^{-T} \Lambda^{\frac{1}{2}}$ , then we obtain  $XX^T = \Psi \widetilde{M}(\omega_i) \Psi^T$ .



Suppose  $F$  is a matrix of dimension  $m \times l$ , take  $\xi^r = F^T \xi$ . Then

$$\begin{aligned} \int_{\omega_i} \tilde{\kappa} |\xi^T \Psi - \xi^{r^T} \Psi^r|^2 &= (\xi^T \Psi - (\xi^r)^T \mathcal{R} \Psi) \widetilde{M}(\omega_i) (\xi^T \Psi - (\xi^r)^T \mathcal{R} \Psi)^T \\ &= \xi^T (I - F \mathcal{R}) \Psi \widetilde{M}(\omega_i) \Psi^T (I - F \mathcal{R})^T \xi \\ &= \xi^T (X - F \mathcal{R} X) (X - F \mathcal{R} X)^T \xi = \|\xi^T (X - F \mathcal{R} X)\|^2. \end{aligned}$$

In the following, we construct a matrix  $F$  that minimizes  $\|\xi^T (X - F \mathcal{R} X)\|$ . Following [19, Lemma 18], we define

$$F = U^{-T} \begin{pmatrix} \mathcal{H}^{(-1)} \\ 0 \end{pmatrix},$$

where  $\mathcal{H}$  and  $\mathcal{S}$  are matrices of dimension  $l \times k$  and  $l \times (m - k)$  defined as,

$$\mathcal{R} U^{-T} = \begin{pmatrix} \mathcal{H} & \mathcal{S} \end{pmatrix}, \quad (17)$$

$$\mathcal{H}^{(-1)} = (\mathcal{H}^T \mathcal{H})^{-1} \mathcal{H}^T. \quad (18)$$

That is,  $\mathcal{H}$  is of rank  $k$  and contains the first  $k$  columns of  $\mathcal{R} U^{-T}$  and  $\mathcal{H}^{(-1)}$  is the pseudo-inverse of  $\mathcal{H}$ .

We obtain

$$\xi^T (X - F \mathcal{R} X) = -\xi^T U^{-T} \left( \begin{pmatrix} \mathcal{H}^{(-1)} \\ 0 \end{pmatrix} \begin{pmatrix} \mathcal{H} & \mathcal{S} \end{pmatrix} - I \right) \Lambda^{\frac{1}{2}}.$$

Furthermore,

$$\|\xi^T (X - F \mathcal{R} X)\| \leq \|\xi^T U^{-T}\| (\|\mathcal{H}^{(-1)} \mathcal{S}^T\| + \|T\|),$$

where  $T$  is defined as

$$\Lambda^{\frac{1}{2}} = \begin{pmatrix} S & 0 \\ 0 & T \end{pmatrix}. \quad (19)$$

Thus, the spectral norm of  $T$  is bounded, that is,  $\|T\| \leq \frac{1}{\lambda_{k+1}}$ . Then, using standard properties of

subordinated norms we have,

$$\|\xi^T(X - F\mathcal{R}X)\| \leq \|\xi^T U^{-T}\| (\|\mathcal{H}^{(-1)}\mathcal{S}T\| + \|T\|) \quad (20)$$

$$\leq \|\xi^T U^{-T}\| (\|\mathcal{H}^{(-1)}\mathcal{S}\| + 1) \|T\| \quad (21)$$

$$\leq \frac{\|\mathcal{H}^{(-1)}\mathcal{S}\| + 1}{\lambda_{k+1}} \|\xi^T \Psi\|_{A(\omega_i)}. \quad (22)$$

Here, to obtain the last step we have used the relation (16) that implies

$$\|\xi^T U^{-T}\| = (\xi^T U^{-T} \cdot (\xi^T U^{-T})^T)^{\frac{1}{2}} = (\xi^T U^{-T} U^{-1} \xi)^{\frac{1}{2}} \quad (23)$$

$$= (\xi^T \Psi A(\omega_i) \Psi^T \xi)^{\frac{1}{2}} = \|\xi^T \Psi\|_{A(\omega_i)}. \quad (24)$$

Hence,

$$\int_{\omega_i} \kappa |\nabla \chi|^2 |\xi^T \Psi - \xi^{r^T} \Psi^r|^2 \leq \left( \frac{\|\mathcal{H}^{(-1)}\mathcal{S}\| + 1}{\lambda_{k+1}} \right)^2 \|\xi^T \Psi\|_{A(\omega_i)}^2.$$

The proof is complete.  $\square$

**Remark 3. Estimate for  $\|\mathcal{H}^{(-1)}\mathcal{S}\|$ .**  $U$  in the Lemma 2 is orthonormal with respect to the  $A(\omega_i)$ -inner product. If  $U$  is an orthonormal matrix itself, then by [19, Lemma 18],  $\|\mathcal{H}^{(-1)}\mathcal{S}\| \leq \sqrt{l}\beta \frac{1}{\lambda_{k+1}^2}$  for some positive number  $\beta$  and given  $k$ . If  $U$  is not orthonormal, then by applying the Gram-Schmidt process to the first  $k$  columns of  $U^{-T}$  (denoted as  $V_1$ ) as well as the rest of the columns of it (denoted as  $V_2$ ), we can obtain non-singular triangular matrices  $D_1$  and  $D_2$ , and  $S_1$  and  $S_2$  with  $S_1^T S_1 = I$  and  $S_2^T S_2 = I$ , such that

$$U^{-T} = \begin{pmatrix} V_1 & V_2 \end{pmatrix}$$

$$V_1 = S_1 D_1, \text{ and } V_2 = S_2 D_2.$$

Then

$$\mathcal{H}^{(-1)}\mathcal{S} = (\mathcal{H}^T \mathcal{H})^{-1} \mathcal{H}^T \mathcal{S} = (V_1^T \mathcal{R}^T \mathcal{R} V_1)^{-1} V_1^T \mathcal{R}^T \mathcal{R} V_2,$$

and using the expressions for  $V_1$  and  $V_2$ , we obtain

$$\mathcal{H}^{(-1)}\mathcal{S} = D_1^{-1} (S_1^T \mathcal{R}^T \mathcal{R} S_1)^{-1} (D_1^{-T} D_1^T) (S_1^T \mathcal{R}^T \mathcal{R} S_2) D_2.$$

Therefore, we have

$$\|\mathcal{H}^{(-1)}\mathcal{S}\| \leq \|D_1^{-1}\| \|(S_1^T \mathcal{R}^T \mathcal{R} S_1)^{-1} S_1^T R^T\| \|\mathcal{R} S_2\| \|D_2\|.$$

Since the entries of  $RS_1$  and  $RS_2$  are i.i.d. Gaussian random variables of zero mean and unit variance, using [19, Lemma 14], we get the estimate,

$$\|\mathcal{H}^{(-1)}\mathcal{S}\| \leq \sqrt{2lm\beta^2\gamma^2 + 1} \|D_1^{-1}\| \|D_2\|$$

with probability not less than

$$1 - \frac{1}{\sqrt{2\pi(l-k+1)}} \left( \frac{e}{(l-k+1)\beta} \right)^{l-k+1} - \frac{1}{4(\gamma^2-1)\sqrt{\pi m\gamma^2}} \left( \frac{2\gamma^2}{e^{\gamma^2-1}} \right)^m,$$

where  $\beta$  and  $\gamma$  are positive real numbers,  $\gamma > 1$ .

Next, we note that the  $i$ -th diagonal elements of  $D_1$  and  $D_2$  are the norms of  $i$ -th columns of  $V_1$  and  $V_2$ . Moreover,  $\|D_1^{-1}\| \|D_2\|$  is the ratio of the largest diagonal element of  $D_2$  and the smallest diagonal element of  $D_1$ . Since  $U^{-T}U^{-1} = \Psi A \Psi^T$  and  $U^{-T}\Lambda U^{-1} = \Psi \widetilde{M} \Psi^T$ , the estimate of  $\|\mathcal{H}^{(-1)}\mathcal{S}\|$  depends on the norms of the columns of  $U^{-T}$  and, thus, depends on the contrast, in general. In the particular case, we assume that  $\Psi A \Psi^T$  is a diagonal matrix with entries  $\lambda_1 \leq \lambda_2 \leq \dots \leq \lambda_n$ . In this case,  $U^{-T} = U^{-1} = \text{diag}(\lambda_1^{-1/2}, \lambda_2^{-1/2}, \dots, \lambda_n^{-1/2})$  and  $D_1 = \text{diag}(\lambda_1^{-1/2}, \lambda_2^{-1/2}, \dots, \lambda_l^{-1/2})$ ,  $D_2 = \text{diag}(\lambda_{l+1}^{-1/2}, \lambda_{l+2}^{-1/2}, \dots, \lambda_n^{-1/2})$ . Then, it is easy to verify that  $\|D_1^{-1}\| \|D_2\| = \lambda_l^{1/2}/\lambda_{l+1}^{1/2}$  in this case. This estimate shows that the error can be sensitive on the choice of the eigenspace that is selected. In GMsFEM, we usually select the most important eigenvalues that are very small (see [10]), thus, in general, a contrast-dependent situation can be avoided.

In Lemma 2, we have derived the approximation of the randomized snapshot space to the full snapshot space locally in each patch  $\omega_i$ . Next, we present the convergence the GMsFEM using randomized snapshots. The snapshots are obtained by multiplying the local snapshots  $\Psi_{\omega_i}^{\text{snap}}$  with the corresponding partition of unity function  $\chi_i$  (as in Eqn. (7)). To simplify notation we denote by  $\Psi$  the full global snapshots (snapshots for all  $\omega_i$ 's) and by  $\Psi^r$  the full randomized snapshots (snapshots for all  $\omega_i$ 's).

**Theorem 4.** Denote by  $\Psi$  the snapshot matrix and by  $\Psi^r$  the randomized snapshot matrix of dimension  $m \times n$  and  $l \times n$ , respectively, and their ranks are  $m$  and  $l$ , respectively.  $\mathcal{R}$  is a matrix with i.i.d. Gaussian random entries and that  $\Psi^r = \mathcal{R}\Psi$ . Suppose  $u_H$  is solved using the offline

space formed using the snapshot matrix  $\Psi^r$ , and  $u$  is the fine-scale solution of Eqn.(1), then we have

$$\int_D \kappa |\nabla(u - u_H)|^2 \preceq \left( \frac{1}{\Lambda_*} + \left( \frac{1}{\Lambda_*} \right)^2 (\|\mathcal{H}^{(-1)}\mathcal{S}\| + 1)^2 \right) \int_D \kappa |\nabla u|^2 + H^2 \int_D f^2, \quad (25)$$

where  $\Lambda_*$  is defined in (36) and  $l < m < n$ .

*Proof.* Denote  $I^{\omega_i}$  and  $I_r^{\omega_i}$  as arbitrary interpolants from the fine-scale to the space spanned by the rows of  $\Psi$  and  $\Psi^r$  on the coarse neighborhood  $\omega_i$ , respectively. Later, we choose a proper interpolant that reduces the error. Taking into account that the GMsFEM solution,  $u_H$ , provides a minimal energy error, we have

$$\begin{aligned} \int_D \kappa |\nabla(u - u_H)|^2 &\preceq \int_D \kappa |\nabla(\sum_i \chi_i(u - I_r^{\omega_i}u))|^2 \\ &\preceq \sum_i \int_{\omega_i} \kappa |\nabla(\chi_i(u - I^{\omega_i}u))|^2 + \int_{\omega_i} \kappa |\nabla(\chi_i(I_r^{\omega_i}u - I^{\omega_i}u))|^2. \end{aligned} \quad (26)$$

Next, we use the inequalities

$$\int_{\omega_i} \kappa \chi_i^2 |\nabla(u - I^{\omega_i}u)|^2 \preceq \int_{\omega_i} \tilde{\kappa} |u - I^{\omega_i}u|^2 + \left| \int_{\omega_i} f \chi_i^2 (u - I^{\omega_i}u) \right|, \quad (27)$$

$$\int_{\omega_i} \kappa \chi_i^2 |\nabla(I_r^{\omega_i}u - I^{\omega_i}u)|^2 \preceq \int_{\omega_i} \tilde{\kappa} |I_r^{\omega_i}u - I^{\omega_i}u|^2, \quad (28)$$

where  $\tilde{\kappa}$  is defined by (9). Here, we have used the inequality (29) in [10]. Using (27) and (28), and we obtain from (26)

$$\begin{aligned} \int_D \kappa |\nabla(u - u_H)|^2 &\preceq \sum_i \int_{\omega_i} \tilde{\kappa} |u - I^{\omega_i}u|^2 + \sum_i \left| \int_{\omega_i} f \chi_i^2 (u - I^{\omega_i}u) \right| \\ &\quad + \sum_i \int_{\omega_i} \tilde{\kappa} |I_r^{\omega_i}u - I^{\omega_i}u|^2. \end{aligned} \quad (29)$$

Selecting a proper interpolant  $I^{\omega_i}$ , we have

$$\int_{\omega_i} \tilde{\kappa} |u - I^{\omega_i}u|^2 \preceq \frac{1}{\lambda_{k+1}^{\omega_i}} \int_{\omega_i} \kappa |\nabla(u - I^{\omega_i}u)|^2, \quad (30)$$

where  $\lambda_{k+1}^{\omega_i}$  is the eigenvalue that the corresponding eigenvector which is not included in the coarse

space. Similarly, we can show that

$$\begin{aligned} \left| \int_{\omega_i} f \chi_i^2 (u - I^{\omega_i} u) \right| &\preceq \int_{\omega_i} \tilde{\kappa}^{-1} f^2 + \int_{\omega_i} \tilde{\kappa} |(u - I^{\omega_i} u)|^2 \\ &\preceq \int_{\omega_i} \tilde{\kappa}^{-1} f^2 + \frac{1}{\lambda_{k+1}^{\omega_i}} \int_{\omega_i} \kappa |\nabla(u - I^{\omega_i} u)|^2. \end{aligned} \quad (31)$$

We note that  $\int_{\omega_i} \tilde{\kappa}^{-1} f^2 \preceq H^2 \int_{\omega_i} f^2$  if  $|\nabla \chi_i| = \mathcal{O}(H^{-1})$ . Combining the above estimates, we have

$$\begin{aligned} \int_D \kappa |\nabla(u - u_H)|^2 &\preceq \sum_i \frac{1}{\lambda_{k+1}^{\omega_i}} \int_{\omega_i} \kappa |\nabla(u - I^{\omega_i} u)|^2 \\ &\quad + \sum_i \int_{\omega_i} \tilde{\kappa}^{-1} f^2 + \sum_i \int_{\omega_i} \tilde{\kappa} |(I_r^{\omega_i} u - I^{\omega_i} u)|^2. \end{aligned} \quad (32)$$

For a fixed vector  $I^{\omega_i} u \in \Psi$ , by Lemma 2, we can get a corresponding vector  $\xi^r \in \Psi^r$ , such that

$$\int_{\omega_i} \tilde{\kappa} |\xi^r - I^{\omega_i} u|^2 \preceq \left( \frac{\|\mathcal{H}^{(-1)}(\omega_i) \mathcal{S}(\omega_i)\| + 1}{\lambda_{k+1}^{\omega_i}} \right)^2 \|I^{\omega_i} u\|_{A(\omega_i)}^2, \quad (33)$$

for some integer  $k$ . For simplicity, we assume that  $\lambda_{k+1}^{\omega_i}$  is the same eigenvector as in the interpolant defined in (30) by selecting the smallest index.

We define  $I_r^{\omega_i} u = \xi^r$ . Thus using Eqns. (32) and (33), we obtain

$$\begin{aligned} \int_D \kappa |\nabla(u - u_H)|^2 &\preceq \max_{\omega_i} \left( \frac{1}{\lambda_{k+1}^{\omega_i}} \right) \int_{\omega_i} \kappa |\nabla u|^2 + \sum_i \int_{\omega_i} \tilde{\kappa}^{-1} f^2 \\ &\quad + \sum_i \left( \frac{\|\mathcal{H}^{(-1)} \mathcal{S}\| + 1}{\lambda_{k+1}^{\omega_i}} \right)^2 \|I^{\omega_i} u\|_{A(\omega_i)}^2 \end{aligned} \quad (34)$$

$$\preceq \left( \frac{1}{\Lambda_*} + \frac{1}{\Lambda_*^2} (\|\mathcal{H}^{(-1)} \mathcal{S}\| + 1)^2 \right) \int \kappa |\nabla u|^2 + \sum_i \int_{\omega_i} \tilde{\kappa}^{-1} f^2, \quad (35)$$

where

$$\Lambda_* = \min_{\omega_i} \lambda_{k+1}^{\omega_i}. \quad (36)$$

Here, we have used the boundedness property of the interpolant in the energy norm [14]. Assuming

$|\nabla \chi_i| = \mathcal{O}(H^{-1})$ , we get

$$\int_D \kappa |\nabla(u - u_H)|^2 \preceq \left( \frac{1}{\Lambda_*} + \frac{1}{\Lambda_*^2} (\|\mathcal{H}^{(-1)} \mathcal{S}\| + 1)^2 \right) \int \kappa |\nabla u|^2 + H^2 \int_D f^2. \quad (37)$$

□

**Remark 5.** *One can improve the error due to GMsFEM discretization by changing the eigenvalue problem (see [9]) and the error will scale as  $\frac{1}{\Lambda_*^q}$ , for a large  $q$  that depends on the size of the oversampled region. In this case, the error due to GMsFEM discretization will scale as  $(1/\Lambda_*)^n$  for some large  $n$ .*

## 6. Conclusions

In this paper, we study the use of randomized boundary conditions to reduce the computational cost in multiscale finite element methods. Local multiscale finite element basis functions are constructed in each coarse patch by computing snapshot vectors and performing local spectral decompositions. The choice of snapshot vectors and the local spectral decomposition is important for achieving a low dimensional coarse spaces that can approximate the solution accurately on a coarse mesh. For example, the use of harmonic functions computed in oversampled regions improves the accuracy. However, the computation of harmonic functions for all possible boundary conditions in each local region is expensive. Therefore, we propose the use of randomized boundary conditions for computing the snapshot vectors. We show that with a few snapshot vectors, we can compute the basis functions that provide an accuracy that is similar to that obtained using all snapshot vectors. We analyze the method and validate our estimates with numerical evidence. Moreover, we discuss approaches that are more accurate compared to randomized snapshot; however, they are more expensive. Finally, we discuss how adaptive computations can be performed efficiently and robustly within the framework of randomized snapshots where multiscale basis functions are added locally in some regions based on an error indicator.

## References

## References

- [1] T. Arbogast, G. Pencheva, M.F. Wheeler, and I. Yotov. A multiscale mortar mixed finite element method. *Multiscale Model. Simul.*, 6(1):319–346 (electronic), 2007.

- [2] I. Babuška and R. Lipton. Optimal Local Approximation Spaces for Generalized Finite Element Methods with Application to Multiscale Problems. *SIAM Multiscale Modeling and Simulation*, 9:373–406, 2011.
- [3] I. Babuška, V. Nistor, and N. Tarfulea. Generalized finite element method for second-order elliptic operators with Dirichlet boundary conditions. *J. Comput. Appl. Math.*, 218:175–183, 2008.
- [4] E.T. Chung, Y. Efendiev, and G. Li. An adaptive GMsFEM for high-contrast flow problems. *Journal of Computational Physics*, 273:54–76, 2014.
- [5] Martin Drohmann, Bernard Haasdonk, and Mario Ohlberger. Reduced basis approximation for nonlinear parametrized evolution equations based on empirical operator interpolation. *SIAM J. Sci. Comput.*, 34(2):A937–A969, 2012.
- [6] L.J. Durlofsky. Numerical calculation of equivalent grid block permeability tensors for heterogeneous porous media. *Water Resour. Res.*, 27:699–708, 1991.
- [7] Y. Efendiev, J. Galvis, and E. Gildin. Local-global multiscale model reduction for flows in highly heterogeneous media. *Journal of Computational Physics*, 231 (24):8100–8113, 2012.
- [8] Y. Efendiev, J. Galvis, and T. Hou. Generalized multiscale finite element methods. *Journal of Computational Physics*, 251:116–135, 2013.
- [9] Y. Efendiev, J. Galvis, G. Li, and M. Presho. Generalized multiscale finite element methods. oversampling strategies. *International Journal for Multiscale Computational Engineering*, *accepted*, 2013.
- [10] Y. Efendiev, J. Galvis, and X.H. Wu. Multiscale finite element methods for high-contrast problems using local spectral basis functions. *Journal of Computational Physics*, 230:937–955, 2011.
- [11] Y. Efendiev and T. Hou. *Multiscale Finite Element Methods: Theory and Applications*. Springer, 2009.
- [12] Y. Efendiev, T. Hou, and X.H. Wu. Convergence of a nonconforming multiscale finite element method. *SIAM J. Numer. Anal.*, 37:888–910, 2000.

- [13] R. Ewing, O. Iliev, R.D. Lazarov, I. Rybak, and J. Willems. A simplified method for upscaling composite materials with high contrast of the conductivity. *SIAM J. Sci. Comput.*, 31(4):2568–2586, 2009.
- [14] J. Galvis and Y. Efendiev. Domain decomposition preconditioners for multiscale flows in high contrast media. *SIAM J. Multiscale Modeling and Simulation*, 8:1461–1483, 2010.
- [15] M. Ghommem, M. Presho, V. M. Calo, and Y. Efendiev. Mode decomposition methods for flows in high-contrast porous media. global-local approach. *Journal of Computational Physics*, 253:226–238.
- [16] N. Halko, P. G. Martinsson, and J. A. Tropp. Finding structure with randomness: probabilistic algorithms for constructing approximate matrix decompositions. *SIAM Rev.*, 53(2):217–288, 2011.
- [17] T. Hou and X.H. Wu. A multiscale finite element method for elliptic problems in composite materials and porous media. *J. Comput. Phys.*, 134:169–189, 1997.
- [18] O.P. Iliev, R.D. Lazarov, and J. Willems. Fast numerical upscaling of heat equation for fibrous materials. *J. Computing and Visualization in Science*, 13(6):275–285, 2010.
- [19] P.G. Martinsson, V. Rokhlin, and M. Tygert. A randomized algorithm for the approximation of matrices. *Technical report, YALEU/DCS/TR-1361*, 2006.
- [20] X.H. Wu, Y. Efendiev, and T.Y. Hou. Analysis of upscaling absolute permeability. *Discrete and Continuous Dynamical Systems, Series B.*, 2:158–204, 2002.